

Hepatoprotective effects of systemic ER activation

Human ER-sensitive genes and NAFLD classification models

Christian Sommerauer & Carlos Gallardo

11 January, 2024

```
# source and library import
source('code/00_helper_functions.R')
library(tidyverse)
library(RColorBrewer)
library(ComplexHeatmap)
library(patchwork)

# color palettes
colPals <- list()
colPals$conditions <- setNames(c('#44AA99', '#117733', '#88CCEE', '#332288', '#DDCC77', '#CC6677', '#AA4499', '#882255'),
  c('CDf', 'HFDf', 'CDm', 'HFDm', 'DPN', 'DIP', 'E2', 'PPT'))
colPals$RdBu <- rev(RColorBrewer::brewer.pal(n=11, name = 'RdBu'))
colPals$UpDown <- setNames(colPals$RdBu[c(10,2)],
  c('up', 'down'))
colPals$clusters <- setNames(c('#E6E6E6', '#B3B3B3', '#8C8C8C', '#4D4D4D'),
  c('1', '2', '3', '4'))
colPals$celltypes <- setNames(c('#B4272F', '#E5462D', '#FFD1D1', '#F4E54C', '#FBAA3E', '#AA654E', '#B58B80', '#6D7AA5',
  '#B3177E', '#CAC1DD', '#67227D', '#36B449', '#82C349', '#A9D265', '#199478', '#95A3A3', '#C4C5C7'),
  c('Cholangiocytes', 'Endothelial cells', 'HsPCs', 'Stromal cells', 'Hepatocytes',
  'Kupffer cells', 'Monocytes & Monocyte-derived cells', 'T cells', 'NK cells',
  'ILC1s', 'B cells', 'cDC1s', 'cDC2s', 'Mig. cDCs', 'pDCs', 'Basophils', 'Neutrophils'))
colPals$inferno <- c('#FCFFA4', '#FCA50A', '#DD513A', '#932667', '#420A68', '#000004')
colPals$models <- setNames(c('#F6B651', '#92A3A5', '#972D22'),
  c('stage', 'nas', 'fibrosis'))
colPals$models_light <- setNames(c('#FAD7A0', '#BFC9CA', '#DF7B71'),
  c('stage', 'nas', 'fibrosis'))
```

Load data

```
# mouse estrogen-sensitive enhancer-gene pairs
mouse_49_ER_genes <- read.table(
  file = 'results/Epigenome_analysis/corr_49genes_76enh_toPlot.txt',
  stringsAsFactors = FALSE,
  sep = '\t',
  header = TRUE) %>%
  dplyr::pull(symbol) %>%
  unique()

# mouse-human orthologs
mouse_human_orthologs <- read.table(
  file = 'data/ensembl_mmus_hsap_dec2021_orthologs.tsv',
  sep = '\t',
  header = TRUE,
  quote = '')

# mouse differentially expressed genes
DEGs <- readRDS('results/bulkRNAseq_mmus_DEGs.rds')

# NAFLD patient cohort
cohort_data <- readRDS("data/bulkRNAseq_human_cohort_data.rds")

cohort_data$Govaere$cpm_filt <- cohort_data$Govaere$cpm %>%
  tibble::rownames_to_column(var = 'gene') %>%
  dplyr::filter(gene %in% mouse_human_orthologs$GeneID_human) %>%
  dplyr::mutate(gene = dplyr::recode(gene, !!!setNames(mouse_human_orthologs$GeneSymbol_human,
    mouse_human_orthologs$GeneID_human))) %>%
  dplyr::filter(!duplicated(gene) & gene != "") %>%
  tibble::column_to_rownames(var = 'gene')
```

Convert estrogen-sensitive genes in mouse to human orthologs

```
# find genes that are expressed in human patients. Cutoff: 0.5 CPM.
human_detectable <- cohort_data$Govaere$cpm_filt %>%
  apply(., 1, median)
human_detectable <- names(human_detectable[human_detectable > 0.5])

# filter ER genes for those expressed in human
human_ER_genes <- mouse_human_orthologs %>%
  dplyr::filter(GeneSymbol_mouse %in% mouse_49_ER_genes) %>%
  dplyr::filter(GeneSymbol_human %in% human_detectable) %>%
  dplyr::filter(!duplicated(GeneSymbol_human)) %>%
  dplyr::pull(GeneSymbol_human)

# export human ER genes
cat(human_ER_genes,
    file = 'results/ER_sensitive_genes.txt',
    sep = '\n')
```

Integrated heatmap of ER-sensitive genes across human NAFLD spectrum

```
human_stage <- cohort_data$Govaere$cpm_filt %>%
  dplyr::filter(rownames(.) %in% human_ER_genes) %>%
  groupTransform(cohort_data$Govaere$meta$Stage, function(x) apply(x,1,median)) %>%
  scaleData(method = 'zscore') %>%
  dplyr::arrange(rownames(.))

human_steatosis <- cohort_data$Govaere$cpm_filt %>%
  dplyr::filter(rownames(.) %in% human_ER_genes) %>%
  groupTransform(cohort_data$Govaere$meta$NAS, function(x) apply(x,1,median)) %>%
  scaleData(method = 'zscore') %>%
  dplyr::arrange(rownames(.))

human_fibrosis <- cohort_data$Govaere$cpm_filt %>%
  dplyr::filter(rownames(.) %in% human_ER_genes) %>%
  groupTransform(cohort_data$Govaere$meta$Fibrosis, function(x) apply(x,1,median)) %>%
  scaleData(method = 'zscore') %>%
  dplyr::arrange(rownames(.))

mouse_log2FC_HFDmVsCDm <- DEGs$filt$CDmVsHFDm %>%
  dplyr::rename(GeneSymbol_mouse = external_gene_name) %>%
  dplyr::inner_join(mouse_human_orthologs, by = 'GeneSymbol_mouse') %>%
  dplyr::filter(GeneSymbol_human %in% human_ER_genes) %>%
  dplyr::filter(!duplicated(GeneSymbol_human)) %>%
  dplyr::select(GeneSymbol_human, log2FoldChange) %>%
  dplyr::mutate(log2FoldChange = log2FoldChange*-1) %>%
  dplyr::rename(HFDmVsCDm = log2FoldChange) %>%
  dplyr::mutate(HFDmVsCDm = factor(ifelse(HFDmVsCDm>0, 'high', 'low'), levels = c('high','low'))) %>%
  tibble::column_to_rownames(var = 'GeneSymbol_human') %>%
  dplyr::arrange(rownames(.))

mouse_log2FC_HFDfVsCDf <- DEGs$filt$CDfVsHFDf %>%
  dplyr::rename(GeneSymbol_mouse = external_gene_name) %>%
  dplyr::inner_join(mouse_human_orthologs, by = 'GeneSymbol_mouse') %>%
  dplyr::filter(GeneSymbol_human %in% human_ER_genes) %>%
  dplyr::filter(!duplicated(GeneSymbol_human)) %>%
  dplyr::select(GeneSymbol_human, log2FoldChange) %>%
  dplyr::mutate(log2FoldChange = log2FoldChange*-1) %>%
  dplyr::rename(HFDfVsCDf = log2FoldChange) %>%
  dplyr::mutate(HFDfVsCDf = ifelse(HFDfVsCDf>0, 'high', 'low')) %>%
  dplyr::add_row(GeneSymbol_human = dplyr::setdiff(rownames(mouse_log2FC_HFDmVsCDm),
                                                    .$GeneSymbol_human),
                HFDfVsCDf = 'unchanged') %>%
  dplyr::mutate(HFDfVsCDf = factor(HFDfVsCDf, levels = c('high','low', 'unchanged'))) %>%
  tibble::column_to_rownames(var = 'GeneSymbol_human') %>%
  dplyr::arrange(rownames(.))

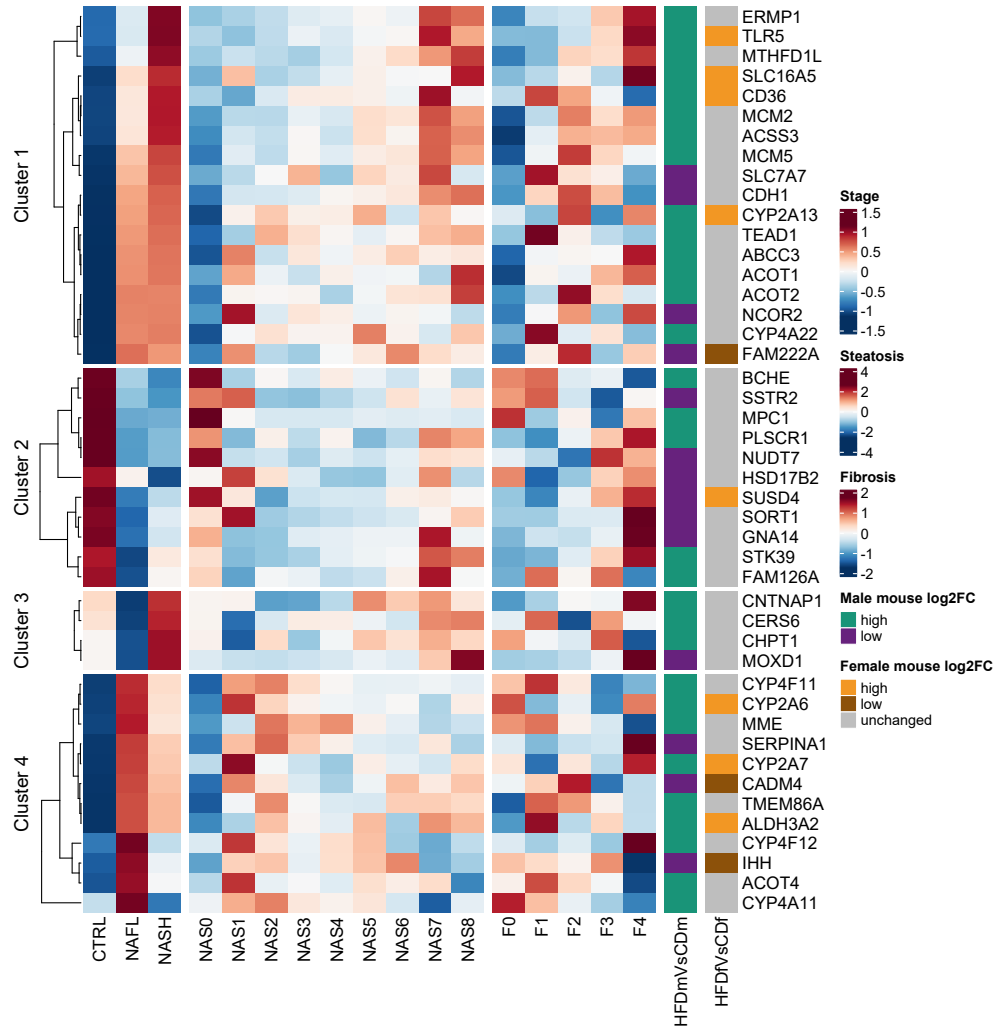
set.seed(30)
clusters_ER_genes <- kmeans(human_stage, centers = 4, iter.max = 100)

Heatmap(human_stage, width = unit(24, "mm"), name = "Stage",
        split = clusters_ER_genes$cluster,
        col = circlize::colorRamp2(breaks=seq(-max(abs(human_stage)), max(abs(human_stage)), length.out=11),
                                   colors=colPals$RdBu),
```

```

row_title = c("Cluster 1", "Cluster 2", "Cluster 3", "Cluster 4"),
cluster_row_slices = FALSE,
column_order=c("CTRL", "NAFL", "NASH")) +
Heatmap(human_steatoisis, width = unit(72, "mm"),name = "Steatoisis",
column_order=paste0('NAS', seq(0,8)),
col = circlize::colorRamp2(breaks=seq(-max(abs(human_steatoisis)), max(abs(human_steatoisis)), length.out=11),
colors=colPals$RdBu)) +
Heatmap(human_fibrosis, width = unit(40, "mm"),name = "Fibrosis",
column_order=paste0('F', seq(0,4)),
col = circlize::colorRamp2(breaks=seq(-max(abs(human_fibrosis)), max(abs(human_fibrosis)), length.out=11),
colors=colPals$RdBu)) +
Heatmap(mouse_log2FC_HFDmVsCDm, width = unit(8, "mm"), name = "Male mouse log2FC",
col = c('#199478', '#67227D')) +
Heatmap(mouse_log2FC_HFDfVsCDf, width = unit(8, "mm"), name = "Female mouse log2FC",
col = c('#F29724', '#8C510A', 'grey'))

```



SessionInfo

```
sessionInfo()
```

```

## R version 4.2.1 (2022-06-23 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## locale:

```

```

## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] grid      stats      graphics  grDevices  utils      datasets  methods
## [8] base
##
## other attached packages:
## [1] patchwork_1.1.2      ComplexHeatmap_2.14.0 RColorBrewer_1.1-3
## [4] lubridate_1.9.2      forcats_1.0.0         stringr_1.5.0
## [7] dplyr_1.1.0          purrr_1.0.1           readr_2.1.4
## [10] tidyr_1.3.0          tibble_3.2.1          ggplot2_3.4.2
## [13] tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
## [1] shape_1.4.6          circlize_0.4.15       GetoptLong_1.0.5
## [4] tidyselect_1.2.0     xfun_0.39             colorspace_2.0-3
## [7] vctrs_0.6.2          generics_0.1.3        stats4_4.2.1
## [10] htmltools_0.5.5      yaml_2.3.7            utf8_1.2.2
## [13] rlang_1.1.1          pillar_1.9.0          glue_1.6.2
## [16] withr_2.5.0          BiocGenerics_0.44.0   matrixStats_0.63.0
## [19] foreach_1.5.2        lifecycle_1.0.3       munsell_0.5.0
## [22] gtable_0.3.3         GlobalOptions_0.1.2   codetools_0.2-19
## [25] evaluate_0.21        knitr_1.43            tzdb_0.4.0
## [28] IRanges_2.32.0       fastmap_1.1.0         Cairo_1.6-0
## [31] doParallel_1.0.17    parallel_4.2.1        fansi_1.0.3
## [34] highr_0.10           scales_1.2.1          S4Vectors_0.36.0
## [37] rjson_0.2.21         hms_1.1.3            png_0.1-8
## [40] digest_0.6.30        stringi_1.7.8         clue_0.3-63
## [43] cli_3.4.1           tools_4.2.1           magrittr_2.0.3
## [46] cluster_2.1.4        crayon_1.5.2          pkgconfig_2.0.3
## [49] timechange_0.2.0     rmarkdown_2.23        rstudioapi_0.15.0
## [52] iterators_1.0.14     R6_2.5.1             compiler_4.2.1

```